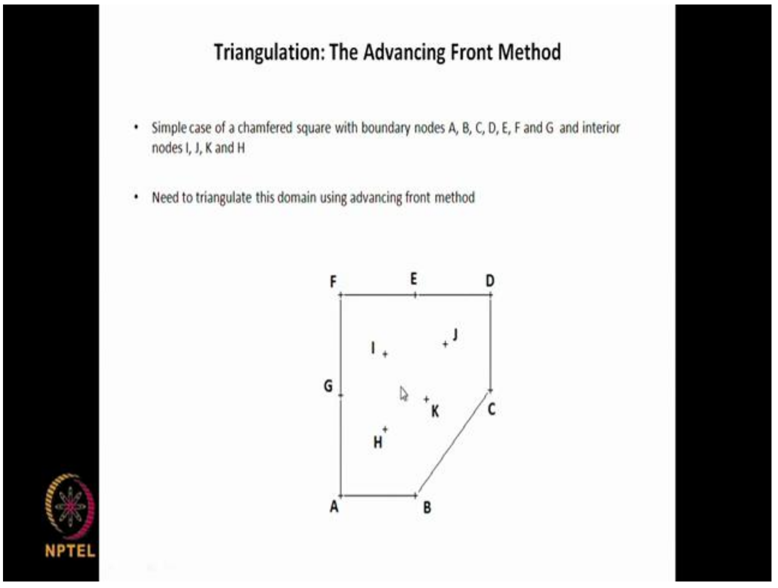
Computational Fluid Dynamics
Prof. Sreenivas Jayanti
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

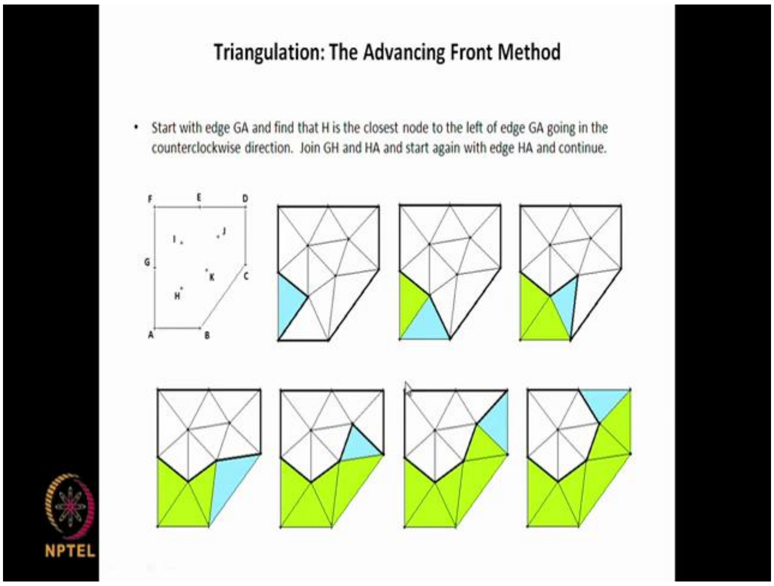
Lecture – 58
The Advancing Front Method continuation

(Refer Slide Time: 00:23)



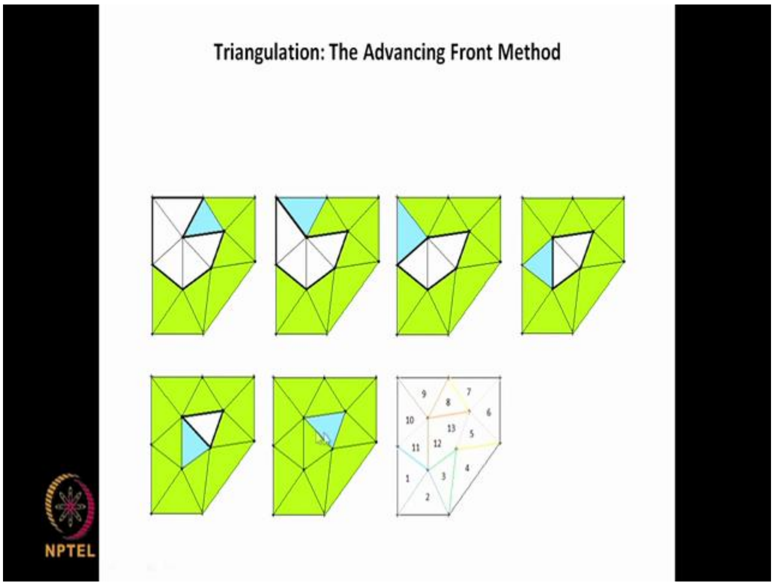
In the last lecture, we have seen how to do triangulation of a domain using the advancing front method. We took this simple shape of chamfered square with eleven nodes including seven boundary nodes and four interior nodes.

(Refer Slide Time: 00:30)



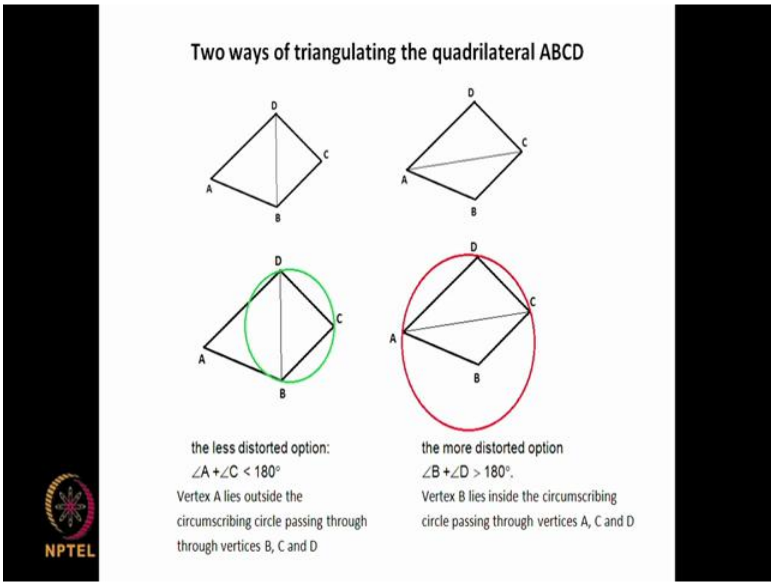
And we started with the edge GA, and then we propagated through these interior nodes, and other nodes seeking each time the node which is to the left and also closes to the edge that is being converted into a triangle. And we are looking at the left, because as we go in the counterclockwise direction looking left means we are going in to interior. So, in that sense, we can progressively go from edge to edge and evolve a front which creates new triangles every time.

(Refer Slide Time: 01:19)



And then we end up with the completely triangulated domain. And we have seen that we are we are made thirteen triangles in the sequence 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13.

(Refer Slide Time: 01:37)

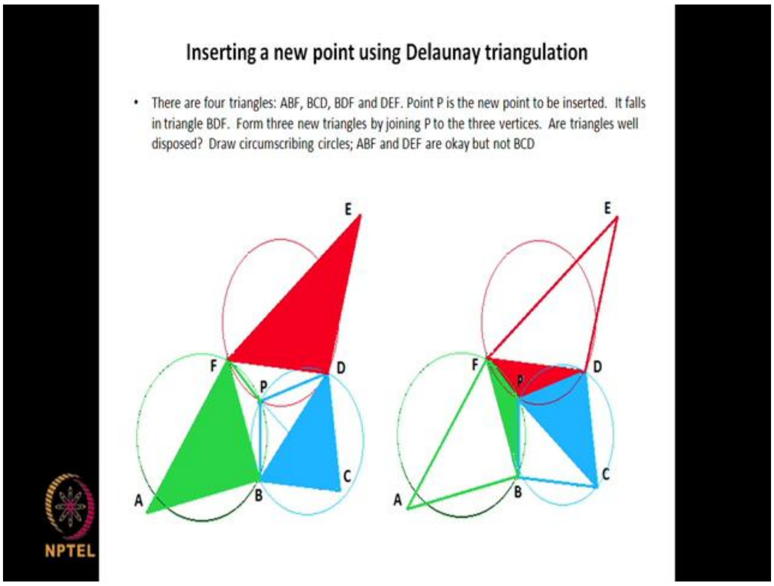


This kind of triangulation as we are discussing earlier does not consider, does not impose the condition that triangles can be made more optimally shaped in the sense of having smaller aspect ratio where we define the aspect ratio as the ratio between the longest side and the shortest side of the triangle. For example, if you take this quadrilateral a b c d you can make it into two triangles by joining the diagonal B D or by joining the diagonal A C. Of these two possibilities, we can see that you have a long edge here and you have a much shorter edge here, so that means, that this particular configuration has larger aspect ratio than this particular configuration. So, this should be preferred as opposed to this.

So, one we have detecting whether we have the less distorted option or the more distorted option is to look at the angle the sum of angles A and C, these are the once which are on opposite of the diagonal B D. And if the sum is less than 180 degrees then joining the BD diagonal will give us a less distorted option. And in this particular case, this is a diagonal the two opposite angles are B and D, and you can see that that is greater than 180 degrees and this gives us an angle more distorted triangulation of the quadrilateral. Another ways to look at a circumscribing circle which passes through points B, C, D of this triangle here, and the one which is the vertex which is left out if that falls outside the circumscribing circle then that is a good way of triangulating, whereas, in this particular case, you have the triangle A, C, D, which is found by joining this diagonal A C. And a circumscribing circle will contain the other vertex which is not part of the circumscribing circle.

If you were to make a circumscribing circle passing through A, B, C again will see that D will be inside that. So, when this happens then you have a more distorted triangulation. So, you can detect this at least by these two criteria and we should try to incorporate this because the more distorted a triangle is the more will be the numerical errors when we do the integration. So, this particular constant is known as the Delaunay triangulation is one in which the triangulation such that if you take any quadrilateral then that quadrilateral is optimally triangulated in the sense that we always take the one option which gives us smaller aspect ratio.

(Refer Slide Time: 04:47)



So, that can be incorporated into grid generation process in this way. So, we are looking at grid generation process, in which we introduce a new node and we try to include that in a triangulation. So, this is some intermediate step will go through the proper method shortly. But what happens when we introduce, and how we can impose this condition of smaller aspect ratio is that to begin with we have four triangles already converted into in the form of triangles ABF - the green one, and then BCD - the blue one, triangle DEF which is a red one, and then the white one DBF or BDF here. And these are the four triangles that we have so far made, and we want into introduce one new node.

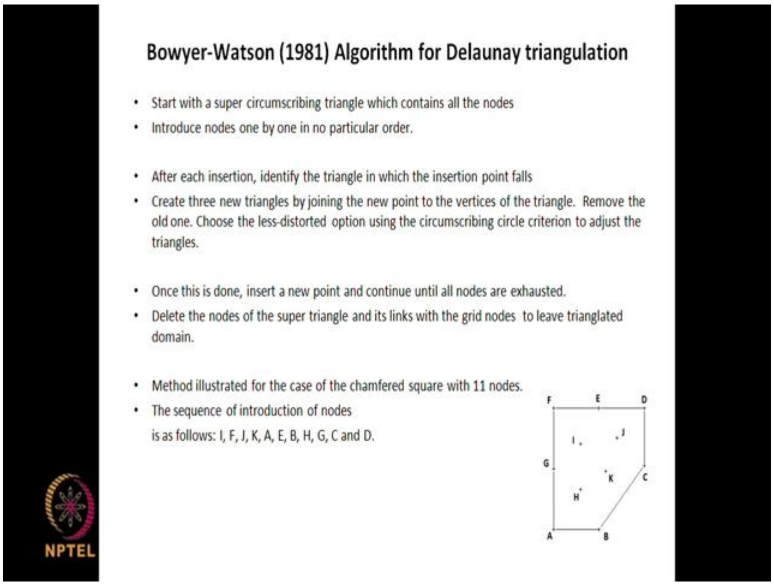
Let us say that new node is P, and algorithm is such that P must fall in one of the existing triangles. So, it may fall in ABF or BCD or DF or BDF. In this case, let say that it falls in BDF and it is somewhere here. So, we can create three new triangles by joining P to the three vertices of the triangle D DFB. So, we have joined PD, PB, and PF. And we have now three more triangles. Now are these things ideally triangulated for example, if i take this quadrilateral consisting of PFED that is the four quadrilateral thing. So, here we have this long triangle option and then we have the short triangle option like this the white portion and the red portion is this triangulated this way or should be triangulated by joining P and E.

So, this is an option, and in order to test that, we can draw circumscribing circle which passes through points FPD and that is the red circle here and you see that the left out vertex E lies outside. So, then that would mean that triangulating by joining DF is better than triangulating by joining PE, and we can see that in this particular case you have a very long edge PE and the short edge PF, so its better this way than in the other way.

Now you come to this green triangle here again you now have a quadrilateral PF, AB you can do it by joining PA or by joining BF, currently BF option. To see if this is good you draw circumscribing circle through BPF and see where vertex a lies and if you do that you see that a is laying outside the circumscribing circle. So, again we can say that joining PA is not as good as joining BF, so we leave this. So, now, you consider this fourth quadrille other quadrilateral that is created here BPDC. And here if you now go by the same argument and draw circumscribing circle, the C falls just inside here so; that means, that this particular configuration it is not so good maybe we should join by putting P and C.

So, now, if you draw join PC here, now you have this triangle the blue triangle and white triangle here. Choose to see if this is really optimal you draw a circumscribing circle which passes through PDC, and then it comes here and you see that b is just outside this is where the border of this is this circle is and B is just outside. So that means that this particular division is better than this division of this quadrilateral. So, at this point, you take any other quadrilateral and you can show that this particular triangulation is better is the best. So, after this now you can introduce new node. So, in this way you can make sure that every time you introduce a node, we have a triangulation in which you have the best option best possible option of a triangulation in terms of the least aspect ratio for all the triangles that exists, so that is Delaunay triangulation concept.

(Refer Slide Time: 09:27)



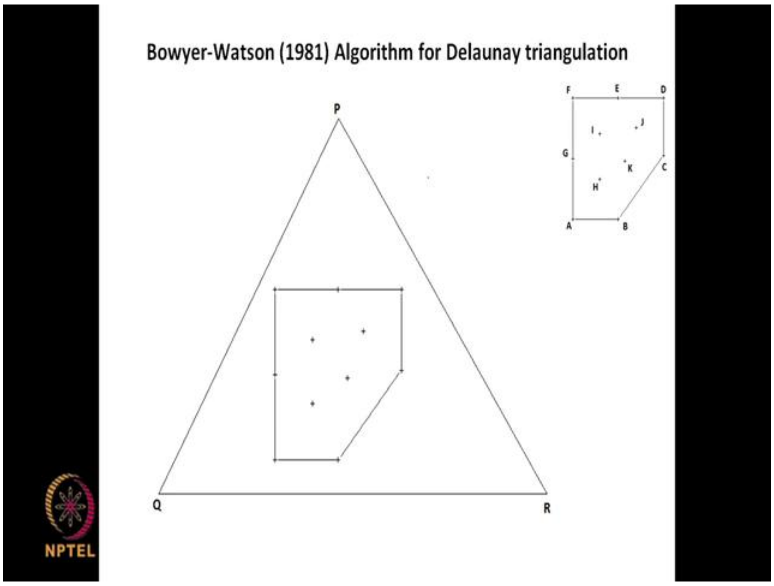
Bowyer and Watson independently proposed in 1981 algorithms to do Delaunay triangulation. And the essence is what is being described here. We will describe first in words and then we go through application of this. So, what we do is that we start with already we have nodalization, so that we have a domain and we have done boundary node nodalization, and we also have the interior nodes. Now it is a question of joining them together in such a way that we get good triangles.

So, we start with a super subscribe circumscribing circle which contains all the nodes, for example, if this is a domain we make sure that all of this domain is captured inside a big triangle. So, we take that triangle and then we introduce nodes one by one in any order, you choose they may the some node order which may be better, but you can do it in any order you like. And at after insertion of each node just as we have done here you introduce a new node you locate which of the excess triangles it falls in and then you join the vertices with this new point here to form new three triangles. And then you delete the old triangle. For example, now triangle BDF is broken up into three new triangles BPD, DPF, and BPF, the old one which is removed and then in place of that you put new three triangles. So, that by the insertion of this point P, you have created a net addition of two triangles.

Now, that is not sufficient then we go through all possible quadrilateral combinations to see using the circumscribing circle criterion whether we have the best option in terms a triangulation and we adjust if necessary and then we move on to the next node, so that is what is the everything here. After each insertion identifies the triangle in which the insertion point falls. Create new three triangles by joining the new point to the vertices of the triangle remove the old one. Choose the less distorted option using the circumscribing circle criterion to adjust the triangles once this is done insert a new point and continue until all the nodes are exhausted.

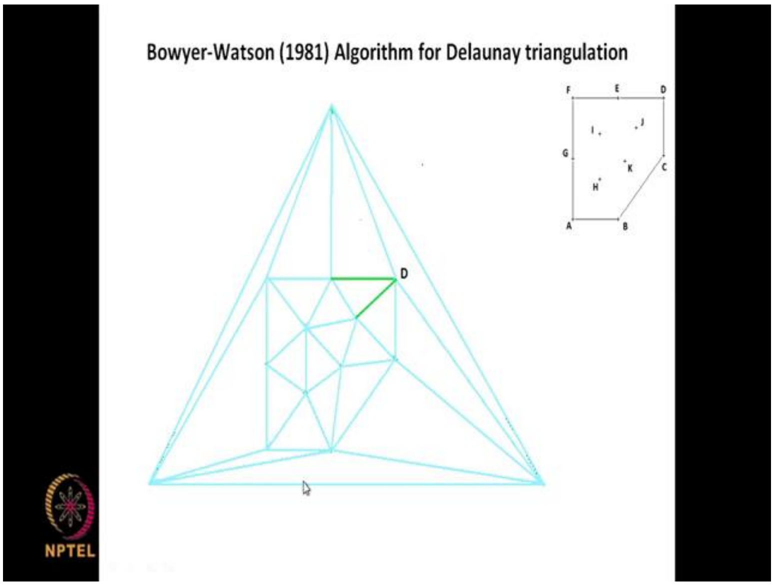
So, once you have introduced all the nodes then you have lots of triangles, and you delete all those triangles in which the nodes are connected to the nodes of the super scribing circle triangle and then that leaves you with a just the triangulated domain. So, this process is illustrated for the same case that we have done earlier for using the advancing front method. So, you have this chamfered square with 11 grid points and we in this algorithm we suppose to introduce this points one by one in no particular order. So, we have chosen to introduce the illustration is for this particular order I, F, J, K, A, E, B, H, G, C and D. So, it just to show that it is not any specific order. So, will go through what we do after each insertion.

(Refer Slide Time: 13:08)



So, we start with choosing the super triangle p q r which inside which this domain would lie. So, the domain is not actually placed here it is just put here for us to visualize better. So, you have an empty triangle inside which you have this domain computation domain with identify nodes. So, this should be large enough to fit that how large it is it does not really matter. So, you have this to begin with you choose the PQR.

(Refer Slide Time: 13:41)



And then you introduce the first point. The first point that we said that we introduces point I here. So, that I would lie here and it; obviously, there is only one triangle. So, inside this, we join the three points here to leave three triangles, and we remove the old triangle PQR in our list of triangles that are already made. So, there is no problem with this, we have three triangles. Now we introduce the next point, which is supposed to be J, I think which is F. So, you have this F point is introduced.

Now, we have to see firstly, into which of this existing triangles this F falls. So, that we do by going through each triangle in the counterclockwise direction going along the edge of each triangle through the counterclockwise direction and c in which triangle F is such that it always lies to the left hand side. For example, we have at this point this is our F we can keep that in mind F is here we start with this triangle counter clock wise means that we going like this as we come from here to here F is to be the right. So, this is F does not obviously belong in this. Now this is the other triangle that we can see and you start from here you go along this and F is again to the right so that means, that this not falling in this. You can also go like this you can go through like this and you can see that F is to the left side. So, it is possible that is inside this, and then you go here and then you come along this edge you find F is to the right side, so that means that it is no longer possible. So, now, you come to this triangle here you start from here, you go along the counterclockwise direction F is to the left, you go along this F is to the left you come along this in the counterclockwise direction and F is still to the left of it. So, it for this triangle the point F is to the left of each of the edges so that means, that F falls inside this triangle. So, now, you join these to make three new triangles then we remove the old one.

So, now at this stage, you have 1, 2, 3, 4, 5 triangles what the three that you originally had plus two new ones which are created. So, now you can introduce the next point which is point j here. So, again first thing is that we had find out into which are the existing triangles j falls. Again we go through the same argument about j being to the left of all the edges as we go in the counterclockwise direction for example, this you go in this direction j is to the left j is to the left here and j is to the left as you go along with this. So, this j falls in this all these other nodes are not yet there, so but it for us to visualize a bit. Always keep in mind what we are trying to do. So, we have this is the thing. So, we join we make new triangles by joining these and then we delete the old one and then we have this here.

Now the question is that are these triangles optimally created. For example, we can take this node this node this node and this node. So, this these form a quadrilateral. Now we have in this we have chosen or at this stage, we have this division here into two triangles into this large triangle here and this triangle. Suppose we were to join i and j and this point, it is possible that gives us a better one. So, we can make this circumscribe and then we circumscribing circle passing through this point, this point and this point and verify for ourselves that this falls outside, therefore that this is the way to do the triangulation here.

So, at this stage, we have added one more point and you are added effectively two new triangles. So, the next point that comes in is point k. So, point k is in this triangle here and so we create points by edges by joining this line, this line and this line here. So, now, we have a quadrilateral we can imagine j this point vertex k and this point here. And we would have chosen to triangulate by joining this, this diagonal here is that the more optimal thing or is it joining this by is more optimal. So, if you draw a circumscribing circle passing through all this things. You can see that this vertex is obviously, always inside so; that means, that joining these two is not a good idea we should be joining these two and that is what we have put here.

So, we have instead of joining these two which are originally there as a result of the test we have decided to join these two, and make triangles this way rather than this particular way. So, this is what we mean by applying that Delaunay triangulation criterion. So, we can see that you have created two triangles here with a total net aspect ratio which is smaller than if you were to join like this. Now if you take this quadrilateral again you can see you have the option of joining like this or like this and this obviously, by the better option. If you take any other quadrilateral, you will see that that is the case.

So, after doing this k, now we introduce the next point which is point a this is the point that we been introduced here. So, we have this we joined a and k here a in this vertex a in this vertex and we create three triangles again we go through the question should we be joining these two to triangulate this quadrilateral that is a k this vertex and this vertex or should we be joining a in this point here. So, when you do that again you can see that this long edge and this is a shorter diagonal and this is better. So, we end up making a triangle like this. So, at this point we are making triangulation like this.

Now if you have this quadrilateral here and now in this quadrilateral point a here this vertex this vertex and this vertex. So, you have these four and we have as it is the proposal is to have these two join together to make two triangles and the other possibilities to join this and this, this is the shorter diagonal and this is the longer diagonal. So, we join this and make e new set of triangles to the point that we want to make is that here at this stage. We have introduced point A and we are created two new diagonals and after applying the Delaunay triangulation criterion, we are joining this and this and not to triangulate the points better. And then after doing this we look at the remaining quadrilaterals and then we make one more change here, so the changes that are made are shown here in green color. So, at this point, we are making change in this diagonal and in this diagonal.

So, now you have a set of triangulation, which is again the best possible thing in this way. So, now, we are ready to insert one more point and this happens to be point e here.

So, this is the point. So, we make this one this one this one. So, now, if you consider this quadrilateral this is the long one which we have joined instead of that if you make the short one that will be better. So, that is what is done in this. So, you have triangles which are made like this. So, in this way, we can we can go through this we can see that in this quadrilateral this is the better one and in this large quadrilateral this is the better one than joining here and so on. We can we can make this kind of calculations verifications to see that this is the best alternative.

Next, we introduce one more point, which is point b. So, point b is somewhere here and this will be like joining this and this and this. So, we can see that there is immediately very thin distorted thing and that we remove by joining this, this is still distorted, but this is better than the other option. So, after this we introduced one more point, we have one two three points to be introduced which one is next point is H. So, at this point, we have to see which of these existing triangles. So, that is this one this one this one this one you have to find out which of these this falls. So, the number of triangles to be tested is increasing and we can see that H falls in these triangles.

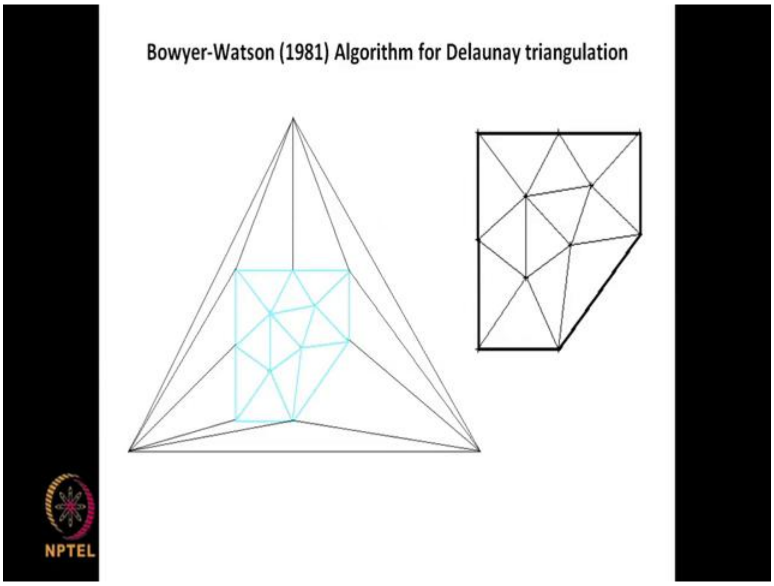
So, we again make join this to create three new triangles removes, the old triangles, and then we look for the quadrilaterals which can be possible alternatively joined, for example, you have this quadrilateral and it is now made into this long diagonal and this so that gives you distorted triangle. So, instead of that if you were to join H B, and then we have two triangles which are better disposed and that is what we have done here. So, we have added two more triangles and we adjusted them a bit, so as to make sure that we have as good triangulation as possible.

Next, we have may be this one now this G is also there. So, this point here again we go through the same process join them like this, and we see that we are creating applying the same criterion to get to make one more change here and then we should be joining this C here. So, C is we have this set like this, you can see that this is a very thin triangle. So, instead of joining these two diagonals, we join like this to make it into this, and then we are also making change here. So, we have this quadrilateral C, this point, this point, this point, those are now being done using this long diagonal; instead of that you were to join this you get better disposed triangles and that is what we have here, two triangles here which are better than join these two and two triangles here which are better than joining these two points.

So, now we have a last point this D, which is to be introduced and obviously, D lies in this. So, we again join these vertices and then we go through the same argument. And you can see now we have this one, this one coming out as superior options to joining in a different way. So, at this stage, we have no more new nodes to be introduced. And we can see that we have recovered this edge, if you were to introduce points here and then join these things this edge would not have been there. But as a result of modification by applying the criterion of good triangulation of quadrilateral, for example, you consider this quadrilateral this thing here we can join it like this or we can join this long diagonal the short diagonal is what should be preferred and that is what we have done here to get this.

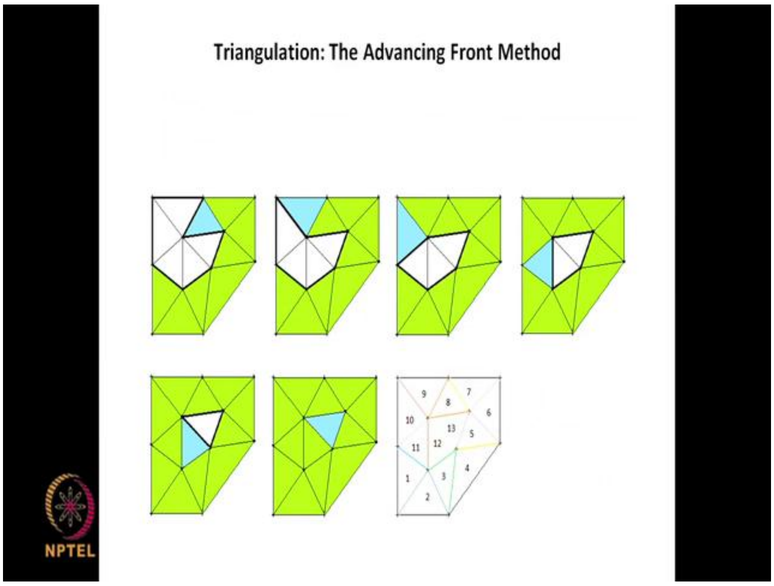
So, the same criterion applied again and again systematically as let us to triangulating all the nodes we started with just one super triangle in which no nodes are there, but this sizes chosen such that it will fit all the things. So, all the nodes are inside this, and then we introduce one by one, does not matter what the order is if you apply that criterion then systematically you will be generating the grids for all of this.

(Refer Slide Time: 27:33)



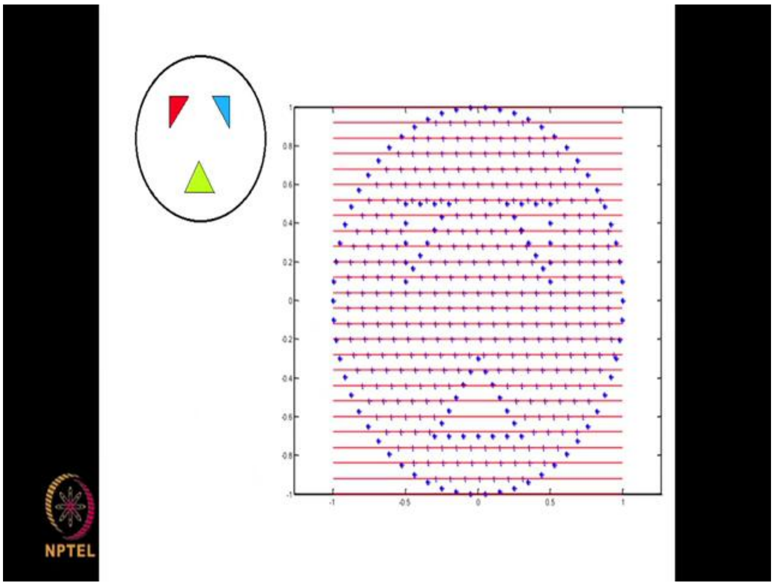
So, at the end, after doing all this things, we have this is the triangulation that you have which includes all joints to the original super diagonal, super triangle and that I marked here as black lines and those are to be removed. Once you removed this point and this point and this point, the three vertices you are left with this as a bit. This is the triangulation, in which you have nice triangles in all sides.

(Refer Slide Time: 28:21)



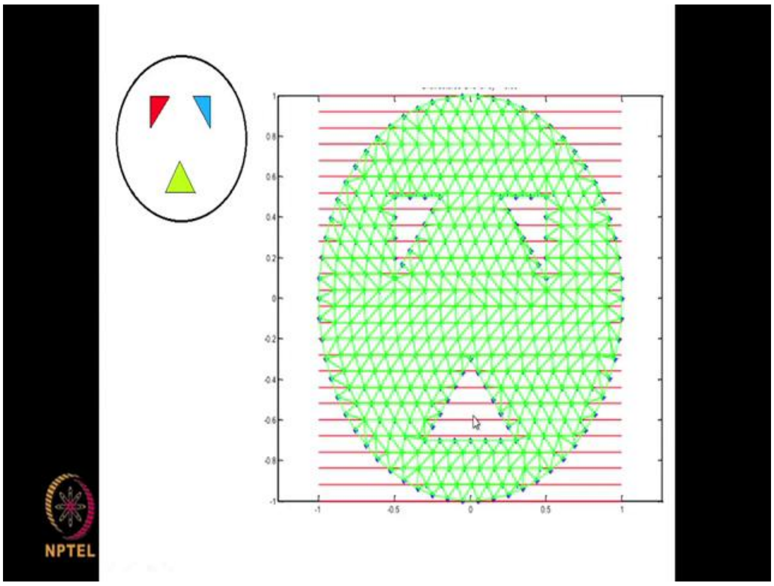
Incidentally this happens to be the same nodilization that we got with this one here with the advancing front method. So, in this particular case, we do not see any overall difference in terms of the final triangulated domain, but we can see that especially in the last step at this point, if you were to join this only like this, and not made the correction then we would not be triangulating this edge, this edge would have been lost. It is only upon the application of that thing we are recovering this edge, which is part of the domain boundary that that this is part of the boundary is not something that is recognized in the process of making this triangulation. It comes out, so there is no distinction between in the boundary nodes and the interior nodes here, we just go ahead and apply the criterion, and then we will be able to recoverable. So, this is the algorithm Delaunay triangulation, and it is very often used, it can be extended to 3-D also, but we are not discussing that.

(Refer Slide Time: 29:56)



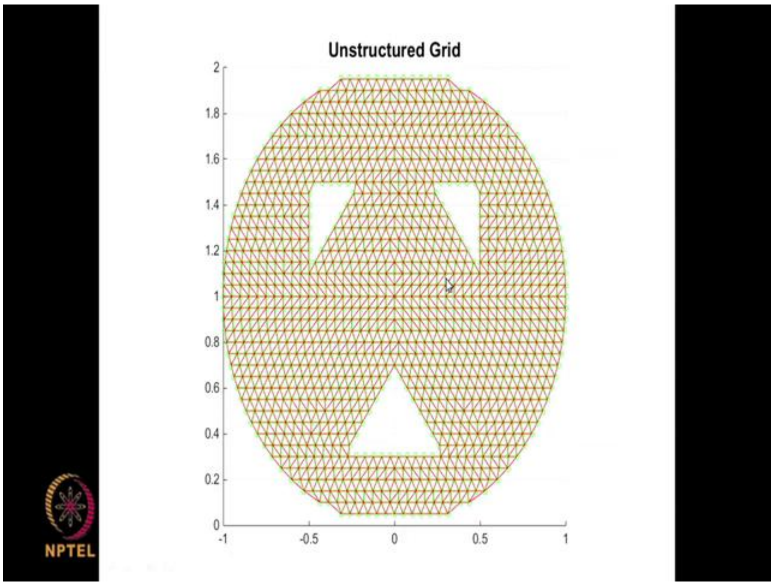
And using this it is possible to make generate diagram for any 2-D shape let me see if I can get one shape we have seen the Delaunay triangulation method, and also the advancing front method. This can be used for a any 2-D problem, for example, you can look at this a thing with a face with a mouth and two eyes like this. And what is the left the white portion is one which is the computational domain so which one student here did triangulation by first domain nodalization by putting these horizontal lines here.

(Refer Slide Time: 30:36)



And then using advancing front method is able to generate a mesh like this.

(Refer Slide Time: 30:41)



One which is slightly smaller size also for the same thing, it is possible to make these kind of shapes any arbitrary shape you can easily generate mesh for this. So, at the end of this, we have identified all the nodes of all this meshes. So, from this, we can go ahead and find the areas, and volumes, and the surface orientations, all those things that you need for a finite volume analysis can be done with this. So, one could do a heat transfer problem in a domain like this with different boundary conditions applied here, and here, here, here, and one could generate the temperature contours in this, so that that kind of problem can be done using the unstructured grid method. So, with this, we can will come to the conclusion of part a of this last module; in the next module, we look at another issue of practical importance which is turbulent flows.

Thank you.